## CLAIMS

What is claimed is:

1. A method for managing memory in a packet switching device comprising the steps of:

(a) managing the memory as a single FIFO when inserting packets; and

(b) managing the memory as a plurality of FIFO queues when removing packets.

2. The method of claim 1 wherein the memory includes a global write pointer (GFWP), a global delete pointer (GFDP), and an empty zone between the GFWP and GFDP.

3. The method of claim 1 wherein the plurality of FIFO are described by tail pointers, head pointers and link pointers.

4. A system for managing memory in a packet switching device wherein the data memory is managed as both a single FIFO, for the purpose of inserting packets, and a plurality of FIFO queues for the purpose of removing packets comprising:

a global FIFO, the global FIFO includes a global FIFO Write Pointer (GFWP) and a Global FIFO Delete Pointer (GFDP), where a plurality of FIFO queues are described by tail pointers, head pointers, and link pointers, each FIFO queue having a tail pointer, head pointer, and link pointer, when a packet is added, it's data is written into the memory sequentially starting at the GFWP, as the packet's data is written the GFWP advances, where

the packets in the memory are stored in a chronological sequential order, starting from above the GFDP (oldest) to just below the GFWP (newest), when a new packet is written into the global FIFO, the plurality of queues database is updated, where the GFDP precedes the GFWP by a preset value, the GFDP advances when a packet's data is written into the memory to maintain the separation, where if the GFDP points to an active start of packet (one that is in a queue) then a packet is removed from the queue for which the packet belongs, where when a packet is read out of a queue, the head pointer for that queue is updated with a link pointer, wherein the data memory is managed as both a single FIFO, for the purpose of inserting packets, and a plurality of FIFO queues for the purpose of removing packets.

5. The system of claim 4 which includes a queue ID memory with an entry per memory section, where a memory section is defined as a block of memory that can have at most one start of packet word (SOPW), where when a SOPW is written the corresponding section queue ID memory is written with the queue ID to which the packet belongs, where when the GFDP advances to a new section, the queue ID is read, and then if the queue is not empty and the head pointer of that queue is equal to the GFDP, then the packet is removed from that queue.

6. The system of claim 4 with a queue ID memory with an entry per memory section, where a memory section is defined as a block of memory that can have at most one start of packet word (SOPW), where when a SOPW is written the corresponding section queue ID memory is written with the queue ID to which the packet belongs and an SOP flag

is set, where when a section is written without an SOPW the SOP flag is cleared, where when the GFDP advances to a new section, the queue ID and SOP flag are read, and then if the SOP flag is set and the queue is not empty and the queue's head pointer is equal to GFDP or is between the GFWP and GFDP, a packet removed operation is scheduled for the queue.

7. A system of claim 4 comprising separately addressable Global FIFO write pointer, Global FIFO delete pointer, packet data memory, section queue ID memory, section queue link memory, head pointers memory, tail section pointers memory, and queue size memory, where a memory section is defined as a block of memory that can have at most one start of packet word (SOPW).